ADAPTING TO RARE REGIMES WITHOUT FORGETTING: LAYERED NEURON PLASTICITY FOR SPATIOTEMPORAL FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Rare but observable regimes, including public holidays and severe weather, account for a disproportionate share of errors in spatiotemporal forecasting. Adapting a deployed forecaster to these regimes with few labeled examples poses a stability–plasticity dilemma: global fine-tuning can disrupt regular-regime behavior, whereas conservative updates fail to capture rare dynamics. We introduce Layered Neuron Plasticity (LNP), a detect–update–restore framework that builds a rare-regime expert while protecting the deployed predictor. Layered Neuron Detection (LND) identifies plastic neuron rows through cross-probe consensus over layer- and projection-specific responses; PNO updates only this support, and Overall Network Restoration (ONR) anchors the adapted rows while re-optimizing their complement on the full training set to restore network-wide coordination. A local second-order analysis links regular-regime forgetting to the support and magnitude of the update, motivating restricted, depth-aware plasticity, while a validation-calibrated inference path conditionally activates the rare expert and corrects residual bias. Across six datasets and ten paired seeds, LNP reduces rare-window MAE relative to the same deployed backbone in every setting while preserving or improving general- and all-window performance. Three matched factorial studies isolate a consistent benefit from restoration, whereas the relative advantage of layered selection varies by setting. Learned masks exhibit increasing plasticity with depth, and independently seeded TimeFilter experiments provide additional evidence of transfer to a second forecasting backbone. Together, these results show that coordinated, layer-aware plasticity enables few-label adaptation to predefined, observable rare regimes without sacrificing regular-regime forecasting performance.

1 INTRODUCTION

Spatiotemporal forecasting supports traffic management, public-transit operation, energy dispatch, and crowd safety. Adaptive graph, Transformer, decomposition, and graph-filtration models have substantially improved average forecasting accuracy Bai et al. (2020); Jiang et al. (2023); Lee & Ko (2024); Shao et al. (2026); Hu et al. (2025). Their training objective is nevertheless dominated by ordinary weekdays and typical operating conditions, which we call the *regular regime*.

The largest deployment errors often occur outside this regime. Public holidays, severe weather, and other identifiable shifts can reshape demand precisely when reliable forecasts matter most. Such *rare regimes* combine distribution shift with very few labeled examples, so their statistical signal is easily overwhelmed by regular data.

Fine-tuning a deployed forecaster on a handful of rare samples is a natural response, but a global update spends those labels across parameters that also encode regular behavior. Figure 1 illustrates the resulting mismatch: holiday traffic follows a shifted pattern while the backbone remains close to the regular commute shape. This is a stability–plasticity dilemma Kirkpatrick et al. (2017); Zenke et al. (2017): the model must learn a visible rare pattern while retaining the much broader regular distribution.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

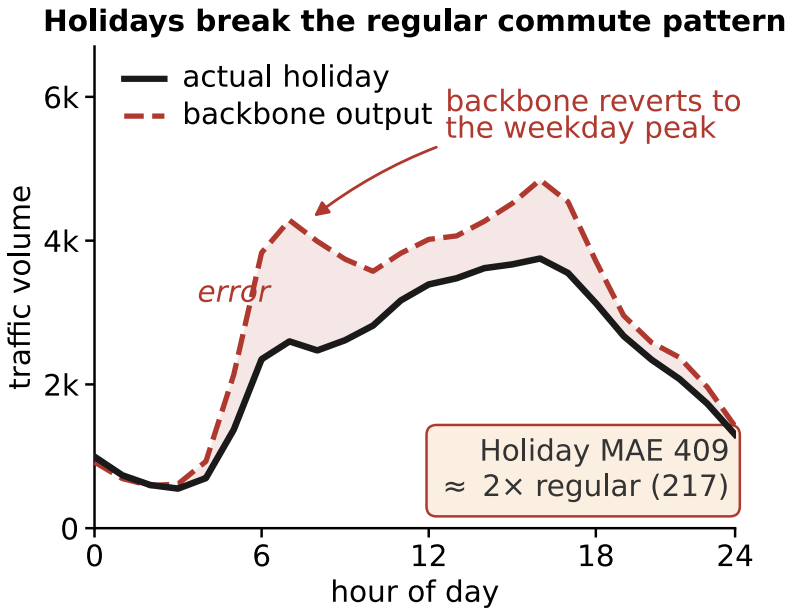

Figure 1: Holiday demand deviates from the regular commute pattern. A backbone optimized for ordinary days reverts toward weekday peaks and misses holiday-specific structure, motivating rare-regime adaptation with explicit regular-regime protection.

PN-Train already detects and fine-tunes neurons associated with infrequent patterns Wang et al. (2025). We investigate whether depth-dependent candidate budgets and complementary-parameter restoration improve this selective adaptation. Our hypothesis is that fixing the adapted rows while adjusting their context can improve the complete predictor. The experiments test this hypothesis separately from the choice of selector.

We implement this idea as **Layered Neuron Plasticity (LNP)**. Layered Neuron Detection (LND) decides where scarce rare labels should be used, PNO applies the masked update, and Overall Network Restoration (ONR) integrates the resulting change into the full predictor. Deployment uses a protected baseline path: a calendar residual head (CaRH) models systematic indicator-conditioned bias, safe conditional fusion (SCF) controls how strongly the rare expert enters the forecast, and calendar residual correction (CRC) removes remaining validation-estimated bias. A local forgetting expansion supplies the design principle behind restricted, layer-aware plasticity and regular-regime protection.

Contributions.

- We study rare-regime adaptation with a small PNO window budget, historical training data, and labeled validation data; a local expansion clarifies why sparse updates alone cannot guarantee retention.
- Building on PN-Train, we combine depth-dependent response selection with ONR, which fixes the adapted rows and optimizes complementary parameters.
- We combine a calibrated baseline, conditional fusion, and validation-residual correction into a protected inference path.
- Across six datasets, ten paired seeds, matched component studies, neuron-mask audits, and a second forecasting backbone, we find consistent rare-regime gains and preserved general performance.

2 Related Work

Pattern-neuron adaptation. PN-Train identifies neurons associated with low-frequency urban patterns and fine-tunes them through a Pattern Neuron Optimizer Wang et al. (2025). LNP retains

this selective-update foundation. Its extensions are depth-dependent candidate budgets, anchored complement restoration, and calibrated inference. Cross-probe intersection and selective fine-tuning are inherited ideas, not independent novelty claims. The uniform-selector and no-ONR controls isolate these extensions within our implementation; they are not a reproduction of the complete original PN-Train protocol.

Spatiotemporal traffic forecasting. Traffic forecasting has progressed from adaptive graph models to Transformer, mixture-of-experts, and decomposition-based architectures. Representative methods include AGCRN Bai et al. (2020), PDFormer Jiang et al. (2023), TESTAM Lee & Ko (2024), STAEformer Liu et al. (2023), HyperD Shao et al. (2026), and TimeFilter Hu et al. (2025); iTransformer Liu et al. (2024) provides a strong general time-series reference. These models target average accuracy under a standard training distribution. LNP addresses the subsequent deployment problem of adapting a trained forecaster with a few labeled rare-regime examples while retaining its regular-regime behavior.

Continual learning and parameter importance. Regularization-based continual learning estimates parameter importance and penalizes changes to important weights. EWC uses Fisher information Kirkpatrick et al. (2017), while Synaptic Intelligence accumulates an online importance trace Zenke et al. (2017); broader surveys study their role in limiting forgetting across task sequences De Lange et al. (2022). Rare and regular regimes instead coexist in one deployed forecaster. LNP couples selective plasticity with a restoration step: LND locates the update support, and ONR re-coordinates the rest of the predictor around that support.

Parameter-efficient adaptation. Adapters and low-rank updates reduce the number of trainable parameters during transfer Houlsby et al. (2019); Hu et al. (2022). Fixed sparse masks choose a persistent parameter subset, for example through Fisher information Sung et al. (2021), and partial-gradient training studies optimization over masked coordinates Mohtashami et al. (2022). LNP makes the sparse support depend on rare-regime responses, layer depth, and agreement across probes, then uses ONR to integrate the PNO update into the original forecaster.

Time-series test-time adaptation provides another nearby comparison. PETSA adapts a frozen forecaster through lightweight input/output calibration and dynamic gating under revealed test targets Medeiros et al. (2025). We study label-limited adaptation to a predefined, observable rare regime. LNP changes a detected subset of backbone neurons, restores their interaction with the predictor, and activates the resulting expert through the observed regime indicator.

Event-aware and non-stationary forecasting. Distribution shift in time series is often handled by changing the input representation, conditioning on external indicators, updating online, or ensembling under drift Zhang et al. (2023). LNP combines event awareness with internal adaptation. The observable indicator determines when the rare expert contributes, while LND+ONR determines how the few labeled examples alter the forecaster. CaRH, SCF, and CRC then preserve and calibrate the baseline path at deployment.

3 Problem and Insight

3.1 Setup

Let f_θ be a forecaster pretrained on historical data, with deployed parameters θ_0. An observable indicator partitions inputs into a regular distribution P_{reg} and a rare distribution P_{rare}. For a forecast input–target pair (x, y) and loss $\ell(f_\theta(x), y)$, define

$$L_q(\theta) = \mathbb{E}_{(x,y)\sim P_q}[\ell(f_\theta(x), y)], \quad q \in \{\mathrm{reg}, \mathrm{rare}\}. \tag{1}$$

Adaptation should reduce L_{rare} while keeping L_{reg} close to its deployed value. Regular-data prevalence alone does not establish stationarity at θ_0. Here a "label" budget counts PNO training windows, each with 12 target steps; historical training and labeled validation data remain available.

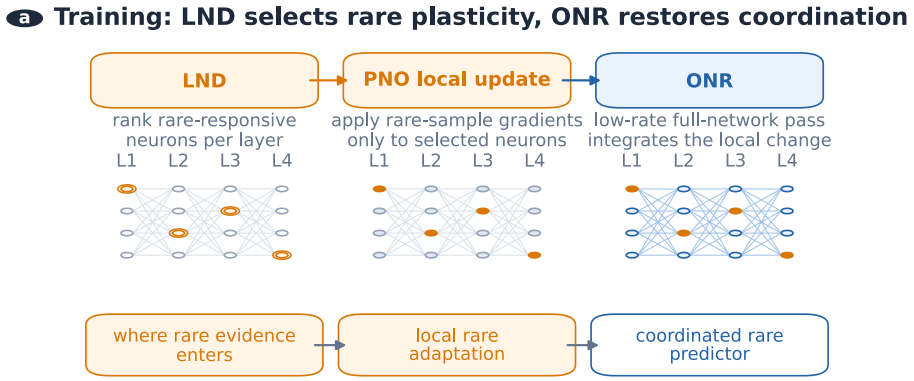

Figure 2: LNP training. LND selects layer-aware neuron rows, PNO updates them with rare labels, and ONR anchors the updated rows while restoring their coordination with the remaining network.

3.2 Naive Adaptation Is Unstable

With only a few rare labels, global fine-tuning has high variance and can move parameters that encode regular behavior. Restricting the number of updated parameters reduces this risk, but restriction alone leaves two questions unresolved: which internal rows should receive the rare signal, and how should the rest of the network use the resulting local change? LNP addresses these questions through selective plasticity followed by full-network restoration.

3.3 A Local Forgetting Expansion

Partition the model parameters into neuron-row groups $\theta = (\theta_1, \ldots, \theta_m)$ and update only groups in S. Thus $\Delta\theta_i = 0$ for $i \notin S$. The following local expansion describes the corresponding regular-regime change.

Proposition 1 (Local regular-regime expansion)**.** *For a twice continuously differentiable loss near θ_0, a sufficiently small update supported on S satisfies*

$$\Delta L_{\mathrm{reg}} = g_S^\top \Delta\theta_S + \tfrac{1}{2}\Delta\theta_S^\top H^{SS}\Delta\theta_S + o(\|\Delta\theta_S\|^2), \tag{2}$$

where $g = \nabla L_{\mathrm{reg}}(\theta_0)$ and $H = \nabla^2 L_{\mathrm{reg}}(\theta_0)$. The quadratic term is at most $\frac{1}{2}\lambda_{\max}(H^{SS})\|\Delta\theta_S\|^2$.

Proof sketch. Taylor expansion gives Eq. equation 2; coordinates outside S contribute zero. The spectral bound follows from the Rayleigh quotient of the symmetric Hessian block. The linear term vanishes only at exact stationarity. □

This smooth-loss argument is a local motivation, not a guarantee for masked MAE, finite updates, ONR, or the final routed predictor. Inspired by importance diagnostics Kirkpatrick et al. (2017), we separately measure

$$Q_{\mathrm{reg}}(\Delta\theta) = \sum_j v_j(\Delta\theta_j)^2, \qquad v_j = \frac{1}{M}\sum_{b=1}^{M}\left[\partial_j \ell_b(\theta_0)\right]^2, \tag{3}$$

where j indexes scalar backbone parameters and ℓ_b is the masked-MAE loss on regular validation batch b. This squared-gradient proxy is not generally Fisher information or a Hessian upper bound. It is used only in a post-PNO diagnostic, not to select neurons or constrain training.

3.4 Selection Principle and Layered Plasticity

LND ranks rare responses and applies layer-dependent candidate budgets followed by cross-probe intersection. It does not estimate a regular-risk budget or solve a Fisher-constrained optimization

problem. Increasing the candidate ratio with depth is an empirical design choice. The resulting depth pattern partly reflects that choice and cannot itself prove a hierarchy of task-specific neurons. Selector ablations, ONR comparisons, and final prediction metrics test the design beyond its mask geometry.

4 METHOD

4.1 OVERVIEW

LNP follows a detect–update–restore training sequence and a protected inference sequence. As shown in Figure 2, **Layered Neuron Detection** (LND) selects a restricted plastic set $S^\star$, PNO updates that set with rare labels, and **Overall Network Restoration** (ONR) holds the updated rows fixed while adjusting their complementary context. Figure 3 shows the deployment path: CaRH calibrates the baseline, SCF gates the LND+ONR rare expert, and CRC corrects the remaining structured residual. LND and ONR are the central adaptation mechanisms; the inference modules preserve their gain across rare and regular windows.

4.2 LAYERED NEURON DETECTION (LND)

Given rare probes $\mathcal{B} = \{x_b\}_{b=1}^{B}$, LND records response $a_{b\ell ci}$ for neuron row i in layer ℓ and projection c. In the STAEformer implementation, this is the signed sum of that channel's activations over batch, time, and node axes (`detect_func=r`); no gradient or Fisher score enters the ranking. Components include temporal/spatial query, key, value, output, and feed-forward projections. LND assigns the candidate budget

$$K_{\ell c} = \lceil r_\ell d_c \rceil , \qquad C_{b\ell c} = \operatorname*{TopK}_{K_{\ell c}} \{a_{b\ell ci} : i = 1, \ldots, d_c\}, \tag{4}$$

where d_c is the number of neuron rows in component c. The final mask is the strict cross-probe consensus

$$S^\star_{\ell c} = \bigcap_{b=1}^{B} C_{b\ell c}, \qquad |S^\star_{\ell c}| \le K_{\ell c}. \tag{5}$$

Here r_ℓ sets the per-probe search budget, while Eq. equation 5 yields a possibly empty realized mask. We use $r_1 < r_2 < r_3$. The intersection retains rows ranked in the candidate set of every probe; regular responses enter only diagnostics and selector controls.

LNP adaptation procedure. Given a pretrained backbone, rare probes, rare labels, and regular training data, one run proceeds as follows:

1. rank neuron rows within every probe, layer, and projection, then retain the per-probe Top-$K_{\ell c}$ candidates;
2. intersect candidate sets across all probes to obtain $S^\star_{\ell c}$ and export the exact masks;
3. run PNO on rare labels while zeroing gradients outside $S^\star$;
4. anchor $S^\star$ and run validation-selected ONR over its complementary coordinates on the full training set; and
5. fit the indicator gate and residual protection on validation data, leaving test targets untouched.

4.3 PNO LOCAL UPDATE

The Pattern Neuron Optimizer (PNO) Wang et al. (2025) fine-tunes the selected rows on rare samples while masking gradients on other backbone coordinates. Each selected index is identified by its layer and projection. A fresh optimizer with zero weight decay avoids off-mask momentum and decay updates; the submitted weight audit checks the resulting invariance.

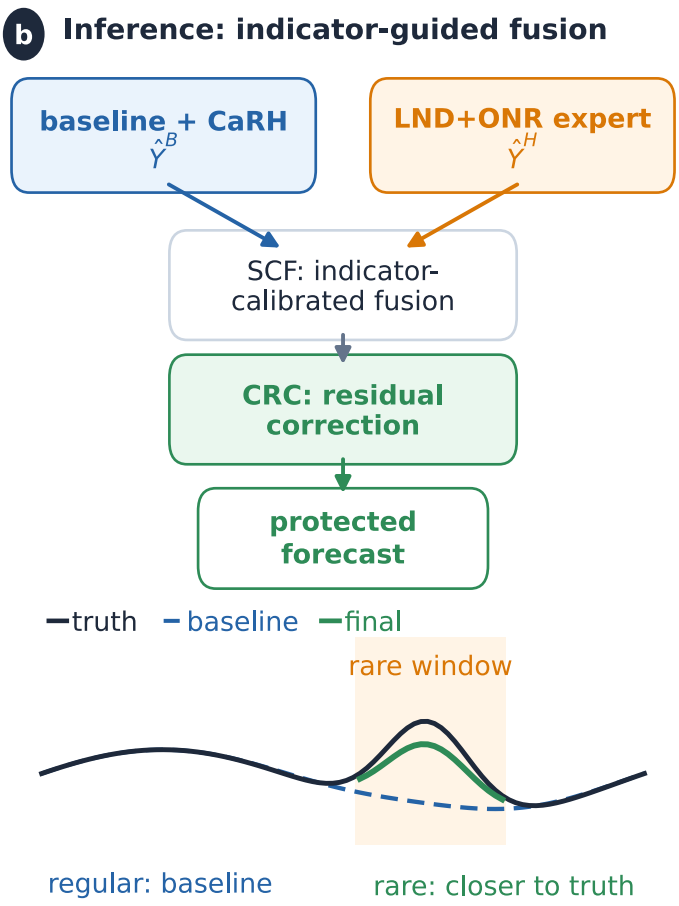

Figure 3: Protected inference: CaRH calibrates the baseline, SCF gates the rare expert, and CRC corrects validation-estimated residual bias.

4.4 Overall Network Restoration (ONR)

PNO changes $S^\star$ without updating its coupled context. ONR restores coordination by optimizing only the complement:

$$\min_{\theta_{\bar{S}}} L_{\text{train}}(\theta_S^{\text{PNO}}, \theta_{\bar{S}}) \quad \text{s.t.} \quad \theta_S = \theta_S^{\text{PNO}}, \tag{6}$$

where L_{train} contains the available historical regular and rare windows. The PNO rows remain bitwise anchored while a low-rate pass adjusts $\bar{S}$. Validation selects among epochs 0–20, including the untouched PNO checkpoint at epoch 0; alternating PNO/ONR is evaluated separately.

4.5 Inference-Time Protection

Let $\hat{Y}^B$ and $\hat{Y}^R$ denote predictions from the deployed backbone and the LND+ONR rare expert. The protection path is

$$\widetilde{Y}^B = \hat{Y}^B + g_\phi(z), \tag{7}$$

$$\widetilde{Y} = (1 - \alpha(z))\widetilde{Y}^B + \alpha(z)\hat{Y}^R, \tag{8}$$

$$\hat{Y} = \widetilde{Y} + \rho_\psi(k). \tag{9}$$

CaRH, g_ϕ, is a zero-initialized residual head conditioned on regime features z. SCF selects $\alpha(z) \in [0, 1]$ on validation data and uses the calibrated baseline outside the rare regime. CRC, ρ_ψ, is a clipped validation-residual table keyed by horizon, hour, day of week, and indicator, with

Dataset	Rare indicator	Labels	Rare MAE	General MAE	All MAE	Rare gain
Metro-Traffic	Holiday	10	409.91 ± 2.75 $\rightarrow 363.51 \pm 14.99$	211.49 ± 2.09 $\rightarrow 205.61 \pm 0.50$	216.67 ± 2.02 $\rightarrow 209.73 \pm 0.83$	46.40 ± 14.28
Pedestrian	Holiday	10	113.16 ± 1.61 $\rightarrow 110.36 \pm 1.21$	81.79 ± 1.04 $\rightarrow 80.45 \pm 0.97$	82.96 ± 1.06 $\rightarrow 81.57 \pm 0.97$	2.80 ± 1.68
Bike-Sharing	Holiday	10	69.63 ± 0.93 $\rightarrow 63.35 \pm 2.50$	66.56 ± 0.99 $\rightarrow 66.56 \pm 0.99$	66.69 ± 0.98 $\rightarrow 66.42 \pm 0.89$	6.28 ± 2.96
Bike-Weather	Severe weather	50	66.93 ± 0.72 $\rightarrow 63.26 \pm 1.13$	61.55 ± 2.25 $\rightarrow 61.55 \pm 2.25$	62.09 ± 2.04 $\rightarrow 61.72 \pm 2.03$	3.67 ± 0.72
MTA-Subway	Holiday	10	172.66 ± 4.82 $\rightarrow 156.41 \pm 7.08$	139.06 ± 8.08 $\rightarrow 106.14 \pm 2.52$	140.26 ± 7.73 $\rightarrow 107.93 \pm 2.49$	16.24 ± 8.07
Appliances-Energy	High humidity	20	35.16 ± 0.71 $\rightarrow 34.12 \pm 0.96$	36.40 ± 0.52 $\rightarrow 32.42 \pm 0.35$	36.36 ± 0.51 $\rightarrow 32.48 \pm 0.34$	1.04 ± 0.62
MTA-Subway†	Holiday	10	$198.46 \rightarrow 190.73$	$136.95 \rightarrow 136.95$	$139.15 \rightarrow 138.87$	$7.74\,[4.99, 11.15]$

Table 1: Backbone→LNP MAE. Rows 1–6 use the STAEformer-style backbone, seeds 42–51, and mean±SD. †TimeFilter on 64-node MTA-Subway hourly ridership, fresh seeds 52–56; brackets give the paired-bootstrap 95% CI. Holidays define MTA rarity; Labels is the adaptation budget. Lower is better.

hierarchical fallback for sparse keys. Validation selects the gate, residual scale, and zero-correction option. These choices do not guarantee unchanged regular forecasts: CaRH and CRC can modify the baseline even when $\alpha = 0$.

5 Experiments

5.1 Experimental Questions

We organize the evaluation around six questions: (RQ1) adaptation under scarce labels; (RQ2) forecasting accuracy against current architectures; (RQ3) transfer across rare signals, domains, and backbones; (RQ4) the roles of LND, ONR, and the protection modules; (RQ5) hyperparameter sensitivity; and (RQ6) the structure of the realized neuron masks.

5.2 Setup

Datasets. Metro-Traffic records hourly westbound I-94 volume at Minnesota DOT ATR station 301 Hogue (2019). Pedestrian contains hourly counts from 48 City of Melbourne sensors City of Melbourne (2024a;b). Bike-Sharing records hourly Capital Bikeshare demand in Washington, DC Fanaee-T (2013). Public holidays define rarity in these three datasets. Bike-Sharing has greater overlap between holiday and regular demand, providing a deliberately weaker calendar-shift case.

Bike-Weather uses the same bike-demand series with `weathersit`≥ 3 as its indicator. MTA-Subway uses hourly ridership at 64 NYC station complexes State of New York (2024). Appliances-Energy predicts household energy use Candanedo et al. (2017), with rarity defined by a training-estimated humidity threshold. These are six dataset–regime settings from five source datasets. Window labels use indicator values over both input and forecast intervals. Holidays are known in advance; weather and humidity results assume the corresponding exogenous values are supplied, rather than demonstrate forecasting with unknown future weather.

Baselines. Architecture controls include DLinear Zeng et al. (2023), iTransformer, HyperD, TimeFilter, AGCRN, PDFormer, TESTAM, and a STAEformer-style backbone Liu et al. (2023). They are local implementations, not exhaustive official-repository reproductions. Matched adaptation controls include protected full fine-tuning, output distillation (LwF) Li & Hoiem (2016), calendar-only calibration, and a protocol-matched PETSA port Medeiros et al. (2025). Earlier replay, LoRA, BitFit, EWC, SI, and MAS diagnostics use a separate protocol and are not pooled with final comparisons.

Protocol. Architecture and sensitivity studies use seeds 42–46 ($n = 5$); same-backbone, matched-control, and factorial results use seeds 42–51 ($n = 10$). LND uses 30 rare validation probes and candidate ratios $(0.50, 0.60, 0.75)$. PNO uses 10 training windows, except 50 on Bike-Weather and 20 on Appliances-Energy, for one masked epoch with zero decay. ONR uses historical training data, including other rare windows, at learning rate 10^{-4}. Thus the PNO budget is not the total number

Method	Metro-Traffic			Pedestrian			Bike-Sharing		
	Rare	General	All	Rare	General	All	Rare	General	All
DLinear (2023)	852.02 ±2.43	1238.97 ±1.46	1228.88 ±1.48	171.84 ±0.21	176.63 ±0.10	176.45 ±0.10	116.06 ±0.44	155.64 ±0.32	153.95 ±0.32
iTransformer (2024)	941.47 ±56.74	824.81 ±75.47	827.86 ±74.70	158.93 ±3.23	145.73 ±3.36	146.22 ±3.33	127.59 ±11.94	111.09 ±16.94	111.80 ±16.59
HyperD (2026)	508.00 ±14.62	264.77 ±2.31	271.12 ±2.26	124.98 ±1.48	97.02 ±1.31	98.06 ±1.30	85.94 ±5.37	65.77 ±1.27	66.63 ±1.16
TimeFilter (2025)	605.66 ±9.19	389.20 ±5.22	394.85 ±5.18	122.71 ±1.68	94.77 ±3.02	95.81 ±2.95	68.20 ±0.93	**60.25** **±0.34**	**60.59** **±0.36**
AGCRN (2020)	498.92 ±10.81	292.43 ±10.38	297.82 ±10.28	111.96 ±3.99	94.26 ±1.96	94.92 ±2.00	65.97 ±2.49	79.27 ±1.08	78.70 ±1.12
PDFormer (2023)	468.54 ±30.10	302.78 ±20.54	307.10 ±20.77	145.08 ±3.10	137.04 ±3.36	137.34 ±3.33	64.72 ±2.50	72.58 ±0.53	72.24 ±0.56
TESTAM (2024)	417.39 ±30.79	228.63 ±7.13	233.55 ±7.58	172.74 ±25.18	141.90 ±15.04	–	**63.80** **±4.19**	70.58 ±3.42	70.29 ±3.32
Backbone (2023)	409.12 ±2.08	211.90 ±1.59	217.05 ±1.50	113.07 ±1.55	81.75 ±1.04	82.92 ±1.06	69.84 ±0.97	66.53 ±0.76	66.67 ±0.77
LNP (ours)	**353.31** **±8.31**	**205.39** **±0.26**	**209.25** **±0.45**	**110.47** **±1.81**	**80.78** **±1.36**	**81.89** **±1.37**	63.88 ±0.47	66.53 ±0.76	66.41 ±0.72

Table 2: Local-implementation MAE over seeds 42–46. LNP uses 10 PNO windows plus historical training and validation data. These are architecture references, not compute-matched adaptation controls. TESTAM/Pedestrian All is not reported because its archived split metrics fail aggregation consistency. Lower is better.

of labeled windows available to the pipeline. Validation selects restoration and calibration; no test target is used for these choices. The mixed criterion is 0.65 all-window plus 0.35 rare MAE on Metro, Pedestrian, and MTA; bike settings use rare MAE and Appliances uses all-window MAE. Tables report mean±sample SD. Paired seed-bootstrap intervals quantify run variability on fixed splits, not uncertainty over independent rare events.

RQ1: Adaptation with scarce labels. With ten Metro PNO windows, protected full fine-tuning, LwF, and calendar-only calibration obtain 391.33, 386.69, and 393.36 rare MAE, compared with 363.51 for LNP. Paired gaps favor LNP in all ten seeds ($p \leq .002$). PETSA reaches 385.55 rare and 211.05 all-window MAE; LNP improves these by 22.03 (95% CI $[13.15, 30.21]$) and 1.32 ($[0.80, 1.78]$), with exact $p = .0039/.0059$. These comparisons support the complete pipeline under the stated data access. They do not isolate a total-compute-matched benefit or establish the superiority of layered selection alone.

RQ2: Main forecasting accuracy. Table 2 compares all models with the same loader, scaler, horizon, split, and validation rule; publication years identify the architectural generation of each control. Full-training architectures provide an absolute-accuracy reference, while LNP adapts the deployed backbone with ten rare labels.

LNP has the lowest mean rare MAE on Metro-Traffic and Pedestrian among the reported local controls. On Bike-Sharing, TESTAM has slightly lower rare MAE, and TimeFilter leads on general and all windows. These rankings are descriptive, not evidence of superiority under exhaustive architecture-specific tuning.

RQ3: Generalization across regimes and models. Table 1 evaluates the same adaptation framework with the dataset-specific settings stated above.

LNP improves rare MAE on all six datasets. The calendar datasets cover one-node traffic, 48-sensor pedestrian flow, one-node bike demand, and 64-station subway ridership; Bike-Weather replaces the holiday indicator with severe weather. Appliances-Energy supplies the non-traffic test: rare/all MAE falls from $35.16/36.36$ to $34.12/32.48$, with positive paired gains in all ten seeds (exact $p = .002$). Bike-Sharing still gains 6.28 rare MAE, but its all-window change is small because rare windows occupy a limited share and their target distribution overlaps more with regular demand, as quantified in the supplement.

The † row tests a local TimeFilter implementation on MTA-Subway with fresh seeds 52–56. Rare/all MAE changes from $198.46/139.15$ to $190.73/138.87$; gains are 7.74 (95% CI $[4.99, 11.15]$) and

0.28 ($[0.18, 0.40]$), with $5/5$ wins. General predictions are exactly the original backbone by routing. All five runs select ONR epoch zero, so this test supports sparse-adaptation portability, not an ONR benefit. Uniform selection reaches rare MAE 190.07 versus layered 190.73; the paired difference is inconclusive ($p = .75$).

RQ4: LND, ONR, and protection mechanisms. A ten-seed factorial holds PNO and inference protection fixed, then crosses layered versus uniform selection with ONR absent versus present. ONR lowers rare MAE by 23.29 on Metro, 6.52 on MTA, and 6.26 on Bike-Weather; Holm-adjusted exact p values are .0078, .0469, and .0059. With ONR, uniform-minus-layered gaps are 9.46, 0.50, and -0.03 ($p = .117/.379/.832$). The selector advantage therefore varies by setting, whereas the ONR gain is consistent. Weight audits confirm that Eq. equation 6 preserves the selected PNO rows.

A separate five-seed Metro component study has full-pipeline rare MAE 359.14, distinct from the architecture and ten-seed protocols. Removing SCF or ONR raises rare MAE by 27.31 and 27.38; removing CRC raises rare/all MAE by $15.85/4.11$, and removing CaRH raises them by $4.76/0.53$. These are within-protocol ablations of the complete prediction path, not effects to be added across protocols.

RQ5: Hyperparameter sensitivity. The supplementary sensitivity plots vary the rare-label budget, layer-wise candidate ratios, and restoration. Rare gains remain positive from 1 to 50 labels and peak at 10 labels; the layered schedule gives the best rare/general balance among the tested ratios. The rare-window gain rises from 10.9% with one label to 13.6% with ten, remains above 9.5% at 30 and 50 labels, and general and all-window gains stay positive throughout. The $(0.50, 0.60, 0.75)$ schedule gives the strongest rare and all-window results, while ONR changes relative rare/general/all MAE from 100 to $91.4/99.5/99.1$. Sequential and alternating schedules are reported in the supplement.

RQ6: Neuron-level structure. The supplementary mask visualizations show the actual response scores and masks consumed by PNO.

Shallow/middle/deep selection fractions are $5.9/15.6/33.5\%$ on Metro and $10.1/19.1/38.5\%$ on MTA. Chance-corrected probe agreement is positive in every seed on Metro, MTA, and Bike-Weather (mean excess $0.202/0.223/0.173$). This measures agreement beyond random candidate overlap, not causal neuron importance. Reversing the depth schedule changes rare MAE by 3.47 on Metro and 0.82 on MTA, with exact $p = .486/.076$. Neither passes a .05 test; depth-wise mask growth partly follows the imposed schedule.

Interpretation and scope. ONR has the more consistent matched effect; evidence for the layered selector is weaker. ONR trains the complement of the sparse mask, so LNP is not parameter-efficient throughout training. Retention is measured for the complete routed predictor, not guaranteed for the rare expert. Evidence covers one second-backbone and one non-traffic setting; two regimes share the bike dataset. Historical labels, labeled validation, and known indicators are required. Future-unknown weather, independent-event uncertainty, direct PN-Train comparison, and matched end-to-end compute remain outside the demonstrated scope.

6 Conclusion

LNP extends pattern-neuron adaptation with layer-dependent selection, anchored restoration, and calibrated inference. Six dataset–regime settings show lower rare MAE and preserved mean regular performance for the complete pipeline. Matched factorials support restoration, but do not establish a consistent layered-selector advantage. These results support adaptation with historical data and known indicators; they do not imply a general no-forgetting guarantee or forecasting robustness to unknown future events.

Reproducibility statement. The anonymous supplement provides the data-processing description, split and seed scopes, training protocol, exact masks, audit outputs, code, and reproduction commands.

AI assistance statement. Generative AI tools assisted with drafting and editing text, checking mathematical exposition, interpreting reported results, providing methodological feedback, and preparing LaTeX artifacts. Responsibility for the scientific claims, experiments, and final content rests with the authors.

REFERENCES

Lei Bai, Lina Yao, Can Li, et al. Adaptive graph convolutional recurrent network for traffic forecasting. In *NeurIPS*, volume 33, pp. 17804–17815, 2020.

Luis M. Candanedo, Véronique Feldheim, and Dominique Deramaix. Data driven prediction models of energy use of appliances in a low-energy house. *Energy and Buildings*, 140:81–97, 2017. doi: 10.1016/j.enbuild.2017.01.083.

City of Melbourne. Pedestrian counting system (counts per hour). DataVic / City of Melbourne open data record, 2024a. URL https://data.melbourne.vic.gov.au/explore/dataset/pedestrian-counting-system-monthly-counts-per-hour/. DataVic package metadata lists license id as other-open.

City of Melbourne. Pedestrian counting system - sensor locations. DataVic / City of Melbourne open data record, 2024b. URL https://data.melbourne.vic.gov.au/explore/dataset/pedestrian-counting-system-sensor-locations/. Sensor metadata table joined to hourly counts by sensor/location identifier.

Matthias De Lange, Rahaf Aljundi, Marc Masana, et al. A continual learning survey: Defying forgetting in classification tasks. *IEEE TPAMI*, 44(7):3366–3385, 2022.

Hadi Fanaee-T. Bike sharing. UCI Machine Learning Repository, 2013. URL https://archive.ics.uci.edu/dataset/275/bike+sharing+dataset. License listed by UCI as Creative Commons Attribution 4.0 International.

John Hogue. Metro interstate traffic volume. UCI Machine Learning Repository, 2019. URL https://archive.ics.uci.edu/dataset/492/metro+interstate+traffic+volume. License listed by UCI as Creative Commons Attribution 4.0 International.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, et al. Parameter-efficient transfer learning for NLP. In *International Conference on Machine Learning (ICML)*, pp. 2790–2799, 2019.

Edward J. Hu, Yelong Shen, Phillip Wallis, et al. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022.

Yifan Hu, Guibin Zhang, Peiyuan Liu, Disen Lan, Naiqi Li, Dawei Cheng, Tao Dai, Shu-Tao Xia, and Shirui Pan. Timefilter: Patch-specific spatial-temporal graph filtration for time series forecasting. In *International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=490VcNtjh7.

Jiawei Jiang, Chengkai Han, Wayne Xin Zhao, and Jingyuan Wang. PDFormer: Propagation delay-aware dynamic long-range transformer for traffic flow prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 4365–4373, 2023. arXiv:2301.07945.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017. doi: 10.1073/pnas.1611835114.

Hyunwook Lee and Sungahn Ko. TESTAM: A time-enhanced spatio-temporal attention model with mixture of experts. In *International Conference on Learning Representations (ICLR)*, 2024. arXiv:2403.02600.

Zhizhong Li and Derek Hoiem. Learning without forgetting. arXiv:1606.09282, 2016. URL https://arxiv.org/abs/1606.09282.

Hangchen Liu, Zheng Dong, Renhe Jiang, et al. Spatio-temporal adaptive embedding makes vanilla transformer sota for traffic forecasting. In *ACM CIKM*, pp. 4125–4129, 2023.

Yong Liu, Tengge Hu, Haoran Zhang, et al. itransformer: Inverted transformers are effective for time series forecasting. In *International Conference on Learning Representations (ICLR)*, 2024.

Heitor Rapela Medeiros, Hossein Sharifi-Noghabi, Gabriel L. Oliveira, and Saghar Irandoust. Accurate parameter-efficient test-time adaptation for time series forecasting. In *ICML Workshop on Test-Time Adaptation: Putting Updates to the Test*, 2025.

Amirkeivan Mohtashami, Martin Jaggi, and Sebastian Stich. Masked training of neural networks with partial gradients. In *Proceedings of the 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pp. 5876–5890, 2022.

Minlan Shao, Zijian Zhang, Yili Wang, Yiwei Dai, Xu Shen, and Xin Wang. HyperD: Hybrid periodicity decoupling framework for traffic forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(18):15689–15697, 2026. doi: 10.1609/aaai.v40i18.38599.

State of New York. MTA subway hourly ridership: 2020–2024. `https://data.ny.gov/Transportation/MTA-Subway-Hourly-Ridership-2020-2024/wujg-7c2s`, 2024. Accessed 2026-07-04.

Yi-Lin Sung, Varun Nair, and Colin Raffel. Training neural networks with fixed sparse masks. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

Chengxin Wang, Yiran Zhao, Shaofeng Cai, and Gary Tan. Investigating pattern neurons in urban time series forecasting. In *International Conference on Learning Representations (ICLR)*, 2025. URL `https://proceedings.iclr.cc/paper_files/paper/2025/hash/1fb38e9c776b5ff18511a284570b15c8-Abstract-Conference.html`.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 11121–11128, 2023. doi: 10.1609/aaai.v37i9.26317.

Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International Conference on Machine Learning (ICML)*, pp. 3987–3995, 2017.

Yifan Zhang, Qingsong Wen, Xue Wang, Weiqi Chen, Liang Sun, Zhang Zhang, Liang Wang, Rong Jin, and Tieniu Tan. Onenet: Enhancing time series forecasting models under concept drift by online ensembling. In *NeurIPS*, 2023.